Semidefinite Relaxations Approach to Polynomial Optimization and One Application

Li Wang

University of California, San Diego

Institute for Computational and Experimental Research in Mathematics

September 9, 2014

Outline

- Introduction: Semidefinite Relaxations for Polynomial Optimization
- Application: SDP Relaxations for Tensor Best Rank-1 Approximations

Outline

- Introduction: Semidefinite Relaxations for Polynomial Optimization
- Application: SDP Relaxations for Tensor Best Rank-1 Approximations

Polynomial Optimization Problem

Consider the following problem:

$$\begin{cases} f_{\min} := \min\limits_{x\in\mathbb{R}^n} \; f(x) \\ \qquad\qquad\quad \text{s.t. } h_1(x) = \cdots = h_{m_1}(x) = 0, \\ \qquad\qquad\qquad\quad g_1(x) \geq 0, \ldots, g_{m_2}(x) \geq 0. \end{cases}$$

This problem is NP-hard even if $f(x)$ is nonconvex quadratic and all constraints are linear.

Standard Method: Lasserre's SDP Relaxation

Sum of Squares (SOS)

A polynomial f is SOS if it is a sum of squares of other polynomials, i.e.,

$$f = \sum_{j=1}^{r} q_j^2.$$

Let f be a degree $2d$ polynomial, f is SOS if and only if there exists

$X \succeq 0$, such that

$$f(x) = [x]_d^T X [x]_d = X \bullet ([x]_d [x]_d^T).$$

Here $[x]_d$ denotes the vector of monomials

$$[x]_d = [1 \quad x_1 \cdots x_n \quad x_1^2 \quad x_1 x_2 \cdots \cdots x_n^d]^T.$$

The length of $[x]_d$ is $N = \binom{n+d}{d}$.

An SDP formulation of Sum of Squares

Define 0/1 constant symmetric matrices C and A_α in the way that

$$[x]_d[x]_d^T = C + \sum_{\deg(x^\alpha)\leq 2d} A_\alpha x^\alpha.$$

If $f(x) = \sum f_\alpha x^\alpha$, then $f(x)$ is SOS if and only if

$$\begin{array}{rcl} C \bullet X & = & f_0, \\ A_\alpha \bullet X & = & f_\alpha, \quad \forall\, \alpha \in \mathbb{N}^n : 0 < |\alpha| \leq 2d \\ X & \succeq & 0 \end{array}$$

$M_d(y)$ is called moment matrix of order d defined as:

$$M_d(y) := C + \sum_{\deg(x^\alpha)\leq 2d} A_\alpha y_\alpha.$$

Lasserre's SDP Relaxation: SOS version

Let $h := (h_1, \cdots, h_{m_1})$, $g := (g_0, g_1, \ldots, g_{m_2})$ where $g_0 = 1$.

The k-th truncated quadratic module generated by (h, g) is defined as

$$Q_k(h,g) := \left\{ \sum_{j=1}^{m_1} \phi_j h_j + \sum_{i=0}^{m_2} \sigma_i g_i \middle| \begin{array}{l} \sigma_i \text{ are SOS}, \ \phi_j \in \mathbb{R}[x], \ \forall\, i,j \\ \deg(\sigma_i g_i) \le 2k, \ \deg(\phi_j h_j) \le 2k \end{array} \right\}.$$

The k-th Lasserre's SOS relaxation (k is also called the relaxation order) is

$$f_k := \max\ \gamma \quad \text{s.t. } f(x) - \gamma \in Q_k(h,g).$$

It is equivalent to an SDP problem.

Lasserre's SDP Relaxation: Moment Version

The dual problem of SOS relaxation is:

$$\begin{cases} f_k^* := \min\limits_{y \in \mathcal{M}_{2k}} \langle f, y \rangle \\ \qquad\qquad \text{s.t. } L_{h_j}^{(k-d_{h_j})}(y) = 0,\ j \in [m_1],\ L_{g_i}^{(k-d_{g_i})}(y) \succeq 0,\ i \in [m_2], \\ \qquad\qquad\quad M_k(y) \succeq 0,\ \langle 1, y \rangle = 1. \end{cases}$$

Here $L_{h_j}^{(k-d_{h_j})}(y)$ and $L_{g_i}^{(k-d_{g_i})}(y)$ are the localizing moment matrix, which is linear in y, i.e.

$$L_{h_j}^{(k-d_{h_j})}(y) := \sum_{\alpha} A_\alpha y_\alpha.$$

Convergence of Lasserre's Hierarchy

Theorem

(Lasserre, 2001) If the archimedean condition holds, as $k \to \infty$*,*
$\lim_{k\to\infty} f_k = \lim_{k\to\infty} f_k^* = f_{\min}$.

Lasserre's hierarchy has *finite convergence* if

$$f_k = f_{\min} \quad \text{for some order } k < \infty.$$

Under the Archimedean Condition, Lasserre's SDP relaxation has finite convergence generically.

Finite Convergence Certificate

Let y^* be an optimizer of moment version of Lasserre's SDP relaxation of order k.
If the flat truncation condition (**FTC**) holds, i.e.,

$$\text{rank } M_{t-d}(y^*) = \text{rank } M_t(y^*).$$

for some integer $t \in [\hat{d}, k]$, and $\hat{d} = \max\{d_f, d\}$, then $f_k^* = f_{\min}$.

The FTC is generically satisfiable if the polynomial optimization problem has finitely many minimizers.

Outline

- Introduction: Semidefinite Relaxations for Polynomial Optimization
- Application: SDP Relaxations for Tensor Best Rank-1 Approximations

What Is Tensor?

A tensor of order m and dimension $(n_1, \ldots, n_m)$ is an array $\mathcal{F}$ that is indexed by integer tuples $(i_1, \ldots, i_m)$ with $1 \le i_j \le n_j$ for $j = 1, \ldots, m$, i.e.,

$$\mathcal{F} = (\mathcal{F}_{i_1,\ldots,i_m})_{1\le i_1 \le n_1,\ldots,1\le i_m \le n_m}.$$

Tensors of order m are called m-tensors.

For example, a given tensor $\mathcal{F} \in \mathbb{R}^{3\times 3\times 3}$:

$$\mathcal{F} = \begin{bmatrix} 1 & 2 & 3 & | & 2 & 5 & 4 & | & 3 & 4 & 2 \\ 3 & 5 & 2 & | & 1 & 4 & 2 & | & 1 & 2 & 3 \\ 4 & 7 & 6 & | & 3 & 9 & 2 & | & 5 & 8 & 2 \end{bmatrix}$$

Here $\mathcal{F}_{123} = 4,\ \mathcal{F}_{121} = 2,\ \mathcal{F}_{322} = 9,\ \mathcal{F}_{312} = 3.$

Symmetric Tensors

A tensor $\mathcal{F} \in \mathbb{R}^{n_1 \times \cdots \times n_m}$ is symmetric if $n_1 = \cdots = n_m$ and

$$\mathcal{F}_{i_1,\ldots,i_m} = \mathcal{F}_{j_1,\ldots,j_m} \quad \text{for all} \quad (i_1, \ldots, i_m) \sim (j_1, \ldots, j_m),$$

where $\sim$ means that $(i_1, \ldots, i_m)$ is a permutation of $(j_1, \ldots, j_m)$.

For example, a given symmetric tensor $\mathcal{F} \in \mathbb{R}^{3\times3\times3}$:

$$\mathcal{F} = \left[\begin{array}{ccc|ccc|ccc} 1 & 2 & 3 & 2 & 4 & 5 & 3 & 5 & 6 \\ 2 & 4 & 5 & 4 & 7 & 8 & 5 & 8 & 9 \\ 3 & 5 & 6 & 5 & 8 & 9 & 6 & 9 & 10 \end{array}\right]$$

$\mathcal{F}_{111} = 1, \mathcal{F}_{121} = \mathcal{F}_{211} = \mathcal{F}_{112} = 2, \cdots, \mathcal{F}_{333} = 10.$

Outer Product and Tensor Decomposition

Given vectors $u^1, \ldots, u^m$, define $u^1 \otimes \cdots \otimes u^m$ as

$$(u^1 \otimes \cdots \otimes u^m)_{i_1,\ldots,i_m} = (u^1)_{i_1} \cdots (u^m)_{i_m}.$$

It is a rank-1 tensor in $\mathbb{R}^{n_1 \times \cdots \times n_m}$.

For every $\mathcal{F}$ of order m, there exist $u^{i,j} \in \mathbb{C}^{n_j}$, such that

$$\mathcal{F} = \sum_{i=1}^{r} u^{i,1} \otimes \cdots \otimes u^{i,m}.$$

The smallest r is the rank of $\mathcal{F}$, and is denoted as $\operatorname{rank}(\mathcal{F})$.

It is hard to get a rank decomposition of $\mathcal{F}$.

Best Rank-1 Approximation for Tensors

Given a tensor $\mathcal{F} \in \mathbb{R}^{n_1 \times \cdots \times n_m}$, a tensor $\mathcal{B}$ is a best rank-1 approximation of $\mathcal{F}$ if it is a minimizer of

$$\begin{cases} \min\limits_{\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_m}} \|\mathcal{F} - \mathcal{X}\|^2 \\ \qquad\qquad \text{s.t.} \quad \operatorname{rank} \mathcal{X} = 1. \end{cases}$$

We can express $\mathcal{B}$ as

$$\mathcal{B} = \alpha \cdot u^1 \otimes \cdots \otimes u^m,$$

for $\alpha \in \mathbb{R}$ and $u^1 \in \mathbb{R}^{n_1}, \cdots, u^m \in \mathbb{R}^{n_m}$.

Tensors and Polynomials

Given a tensor $\mathcal{F} \in \mathbb{R}^{n_1 \times \cdots \times n_m}$, define the polynomial

$$F(x^1, \ldots, x^m) := \sum_{i_1, \ldots, i_m} \mathcal{F}_{i_1, \ldots, i_m} (x^1)_{i_1} \cdots (x^m)_{i_m},$$

Note:

- $F(x^1, \ldots, x^m)$ is homogenous in each vector $x^i \in \mathbb{R}^{n_i}$;
- $F(x^1, \ldots, x^m)$ is multi-linear.

Characterization of Rank-1 Approximation

Theorem: (De Lathauwer, De Moor and Vandewalle, 2000)
For a tensor $\mathcal{F}$, the rank-1 approximation problem is equivalent to

$$\begin{cases} \max \ |F(x^1, \ldots, x^m)| \\ \ \text{s.t. } \|x^1\|_2 = \cdots = \|x^m\|_2 = 1, \end{cases}$$

that is, $\mathcal{B}$ is a best rank-1 approximation for $\mathcal{F}$ if and only if

$$\mathcal{B} = \lambda \cdot (u^1 \otimes \cdots \otimes u^m),$$

where $(u^1, \ldots, u^m)$ is a global maximizer of the above problem and $\lambda = F(u^1, \ldots, u^m)$.

Rank-1 Approximation For Symmetric Tensors

Given a symmetric $\mathcal{F}$, $\mathcal{B}$ is a best rank-1 approximation of $\mathcal{F}$ if

$$\mathcal{B} = \lambda \cdot (u \otimes \cdots \otimes u) \quad \text{and} \quad \lambda = f(u),$$

where u is the global maximizer of:

$$\max_{x \in \mathbb{R}^n} \quad |f(x)| \quad \text{s.t.} \quad \|x\|_2 = 1,$$

$$f(x) = \sum_{1 \le i_1, \ldots, i_m \le n} \mathcal{F}_{i_1, \cdots, i_m} (x)_{i_1} \cdots (x)_{i_m}.$$

So, we need to solve two polynomial optimization problems:

$$\text{(I)} \quad \max_{x \in \mathbb{S}^{n-1}} f(x), \qquad \text{(II)} \quad \max_{x \in \mathbb{S}^{n-1}} -f(x).$$

Semidefinite Relaxation (I): Even Order $m = 2d$

Consider the following optimization problem:

$$\begin{cases} \max & f(x) = \sum_{\alpha \in \mathbb{N}_m^n} f_\alpha x_1^{\alpha_1} \cdots x_n^{\alpha_n} \\ \text{s.t.} & (x^T x)^d = \sum_{\alpha \in \mathbb{N}_m^n} g_\alpha x_1^{\alpha_1} \cdots x_n^{\alpha_n} = 1, \\ & [x^d][x^d]^T = \sum_{\alpha \in \mathbb{N}_m^n} A_\alpha x_1^{\alpha_1} \cdots x_n^{\alpha_n} \succeq 0, \end{cases}$$

where $[x^d]$ is the monomial vector:

$$[x^d] = \begin{bmatrix} x_1^d & x_1^{d-1} x_2 & \cdots & x_1^{d-1} x_n & \cdots & x_n^d \end{bmatrix}^T,$$

$$\mathbb{N}_m^n = \{(\alpha_1, \cdots, \alpha_n) \in \mathbb{N}^n \mid \alpha_1 + \cdots + \alpha_n = m\}.$$

Semidefinite Relaxation (II): Even Order $m = 2d$

Replace each monomial x^α by a variable y_α and get :

$$\begin{cases} \max\limits_{y \in \mathbb{R}^{\mathbb{N}_m^n}} \quad \sum\limits_{\alpha \in \mathbb{N}_m^n} f_\alpha y_\alpha \\ \text{s.t. } \langle g, y \rangle := \sum\limits_{\alpha \in \mathbb{N}_m^n} g_\alpha y_\alpha = 1, \\ M(y) := \sum\limits_{\alpha \in \mathbb{N}_m^n} A_\alpha y_\alpha \succeq 0. \end{cases}$$

This is an SDP problem. It can be solved numerically, e.g.,

- Interior point methods (small to moderate sizes);
- Regularization methods (large sizes).

Semidefinite Relaxation (III): Even Order $m = 2d$

Let y^* be the maximizer of SDP problem:

$$\begin{cases} \max\limits_{y \in \mathbb{R}^{\mathbb{N}_m^n}} & \sum\limits_{\alpha \in \mathbb{N}_m^n} f_\alpha y_\alpha \\ \text{s.t.} & \sum\limits_{\alpha \in \mathbb{N}_m^n} g_\alpha y_\alpha = 1, \\ & M(y) \succeq 0. \end{cases}$$

If rank $M(y^*) = 1$, there exists $u^* \in \mathbb{S}^{n-1}$ such that

$$y^* = [(u^*)^m] \quad \text{and} \quad \|u^*\|_2 = 1,$$

and u^* is a global maximizer of

$$\max \quad f(x) \quad \text{s.t.} \quad \|x\|_2 = 1.$$

$$\text{So} \qquad \mathcal{B} = f(u^*) \cdot (u^* \otimes \cdots \otimes u^*)$$

is a best rank-1 approximation.

Semidefinite Relaxation (IV): Even Order $m = 2d$

Let y^* be the solution of SDP relaxation. If rank $M(y^*) > 1$, we are not guaranteed to get best rank-1 approximation.

In this case, choose $\hat{u}$ as: (for some proper chosen $s \in [1, n]$)

$$\tilde{u} = (y^*_{(2d-1)e_s+e_1}, \dots, y^*_{(2d-1)e_s+e_n}), \quad \hat{u} = \tilde{u}/\|\tilde{u}\|.$$

Then we use local optimization method to solve

$$\max_{x \in \mathbb{R}^n} \quad f(x) \quad \text{s.t.} \quad \|x\|_2 = 1,$$

by using $\hat{u}$ as a starting point.

Similarly, we repeat the above process to solve the problem

$$\max \ -f(x) \quad \text{s.t.} \quad x^T x = 1.$$

Algorithm: Rank-1 approximations for even symmetric tensors

Step 1: Solve the semidefinite relaxations of:

$$\text{(I)} \quad \max_{x \in \mathbb{S}^{n-1}} f(x) \quad \text{and} \quad \text{(II)} \quad \max_{x \in \mathbb{S}^{n-1}} -f(x).$$

Step 2: Let y^* and z^* be the solutions of SDP relaxations for (I)-(II), and choose $\tilde{u}, \ \tilde{v}$ from the SDP solutions;

Step 3: If rank $M(y^*) = 1$ and rank $M(z^*) = 1$, we get best rank-1 approximation;

Step 4: If $\max\{\text{rank } M(y^*), \text{rank } M(z^*)\} > 1$, then apply a local optimization method to improve the solutions.

Measure of Approximation Quality

- **Measurement**:
  SDP relaxation is tight or not, we get an upper bound:

$$f_{\mathsf{ubd}} := \max\{|f_{\max}^{\mathsf{sdp}}|, |f_{\min}^{\mathsf{sdp}}|\}.$$

  The error

$$\texttt{aprxerr} := \big||f(u)| - f_{\mathsf{ubd}}\big| / \max\{1, f_{\mathsf{ubd}}\}$$

  is a measure of the approximation quality.

- **Local Optimization Method**:
  Use Matlab optimization ToolBox: **fmincon** to improve the solution if SDP relaxation is not tight.

Example 1

(Kolda and Mayo, 2011) (Zhang et al., 2012)

Consider symmetric tensor $\mathcal{F} \in \mathbb{R}^{3\times3\times3}$:

$$\mathcal{F}_{111} = -0.1281, \mathcal{F}_{112} = 0.0516, \mathcal{F}_{113} = -0.0954, \mathcal{F}_{122} = -0.1958, \mathcal{F}_{123} = -0.1790,$$
$$\mathcal{F}_{133} = -0.2676, \mathcal{F}_{222} = 0.3251, \mathcal{F}_{223} = \ \ 0.2513, \mathcal{F}_{233} = \ \ 0.1773, \mathcal{F}_{333} = \ \ 0.0338.$$

By semidefinite relaxation method, we get the rank-1 tensor $\mathcal{B} = \lambda \cdot u^{\otimes 3}$ with

$$\lambda = 0.8730, \quad u = (-0.3921, 0.7249, 0.5664).$$

The $M(y^*)$ has rank one, so $\lambda \cdot u^{\otimes 3}$ is a best rank-1 approximation. The error `aprxerr`=1.2e-7.

Example 2: Symmetric Random Example

We generate symmetric tensor $\mathcal{F} \in \mathbb{R}^{n\times\cdots\times n}$ of order m, with each entry being a random variable obeying Gaussian distribution (`randn` in Matlab).

(n,m)	(N,M)	time (min,med,max)			aprxerr (min,med,max)
(10,3)	(66,1000)	0:00:01	0:00:01	0:00:03	(7.9e-9, 4.5e-8, 2.9e-6)
(20,3)	(231,10625)	0:00:03	0:00:08	0:00:13	(2.4e-9, 3.6e-7, 4.3e-6)
(30,3)	(496,46375)	0:01:14	0:01:29	0:02:01	(9.1e-9, 7.4e-7, 1.4e-5)
(40,3)	(861,135750)	0:06:32	0:10:04	0:13:09	(1.3e-9, 4.6e-6, 2.3e-3)
(50,3)	(1326,316250)	0:12:39	0:13:34	0:14:01	(3.2e-9, 1.3e-6, 2.0e-3)
(15,4)	(120,3060)	0:00:01	0:00:03	0:00:04	(4.0e-9, 1.1e-7, 1.3e-6)
(20,4)	(210,8854)	0:00:52	0:01:09	0:01:25	(1.2e-8, 1.8e-7, 6.3e-3)
(25,4)	(325,20475)	0:00:30	0:00:35	0:00:56	(4.7e-9, 1.3e-7, 1.0e-5)
(30,4)	(465,40919)	0:06:03	0:07:36	0:09:31	(1.2e-8, 1.1e-6, 9.6e-4)
(35,4)	(630,73815)	0:02:46	0:04:57	0:06:54	(4.1e-8, 1.6e-7, 7.4e-3)
(10,5)	(286,8007)	0:00:08	0:00:14	0:00:17	(4.3e-8, 4.1e-7, 4.1e-6)
(15,5)	(816,54263)	0:03:46	0:03:58	0:07:24	(4.4e-8, 2.5e-6, 1.1e-3)
(20,5)	(1771,230229)	0:28:14	0:30:30	0:43:27	(4.7e-7, 3.7e-6, 5.7e-6)
(10,6)	(220,5004)	0:00:11	0:00:14	0:00:20	(1.3e-7, 6.4e-7, 3.5e-2)
(15,6)	(680,38759)	0:03:14	0:04:19	0:04:53	(4.8e-8, 2.5e-3, 4.9e-2)
(20,6)	(1540,177099)	0:39:28	0:45:39	0:54:59	(2.8e-8, 6.6e-5, 1.0e-2)

Computer Information: Matlab 7.10 on a Dell 64-bit Linux Desktop running CentOS (5.6) with 8GB memory and Intel(R) Core(TM) i7 CPU 860 2.8GHz.

Thank you!